Feedforward-Output Tracking Regulation Control for Human-in-the-Loop Camera Systems

Rares Stanciu, and Paul Y. Oh
Drexel University, Philadelphia PA
Email: ris22@drexel.edu, paul@coe.drexel.edu

Abstract

Platforms like gantries, booms, aircraft and submersibles are equipped with cameras. People teleoperate such platforms to capture desired views of a scene or a target. To avoid collisions with the environment and occluded views, such platforms are often equipped with redundant degrees-of-freedom. The operator must manually coordinate multiple degrees-of-freedom in order to get desired views. Tracking moving targets becomes especially tedious and often requires several highly skilled operators. Visual-servoing some degrees-of-freedom may reduce operator burden and improve tracking performance. This paper builds upon previous successes with human-in-the-loop visual-servoing by marrying the platform's dynamics with computer vision. A broadcast boom is modelled and both simulation and experimental tests contrasting trained and untrained camera operators are presented.

1 Introduction

Human-in-the-loop systems involve an operator who manipulates a device for desired tasks based on feedback from the device and environment. For example, devices like rovers, gantries, and aircraft often possess a video camera where the task is to maneuver the vehicle and position the camera to obtain desired fields-of-view. Such tasks have applications in areas like broadcasting, inspection and exploration. Such device-mounted camera systems often possess many degrees-of-freedom (DOF) because it is important to capture as many fields-of-view as possible. To overcome joint limits, avoid collisions and ensure occlusion-free views, these devices are typically equipped with redundant DOF. Tracking moving subjects with such systems is a challenging task because it requires a well trained operator who must manually coordinate multiple joints. Tracking performance becomes limited to how quickly the operator can manipulate redundant DOF. Figure 1 shows a typical broadcast boom and pan-tilt camera head. Here, the operator can push and steer the dolly, as well as boom, pan and tilt the camera. The particular interest here is to apply *visual-servoing* to augment an operator's ability to track moving targets; computer vision is used to control some DOF so that the operator has fewer DOF to manipulate.

Figure 1 depicts the experimental platform. It was constructed to capture data (image frames and joint encoders), implement controllers and assess performance. Hardware includes a 266 MHz PC, pan-tilt DC motor system and quadrature encoders. This system consists of a gimbaled boom which is mounted on a four-wheel dolly. On the top of the boom, a motorized pan-tilt head rotates a color camera and the framegrabber captures real-time image data. The boom pivots on the steerable dolly to sweep the camera horizontally and vertically. In previous work, proportional control [8] and partitioning [7] have been designed and tested. Here, the pan-tilt motors were visually servoed in order to keep a moving target centered in the camera's field-of-view despite boom or dolly motions. The net effect is what the authors call *human-in-the-loop visual servoing* – the operator focuses on safely manipulating the boom and dolly while computer-control automatically servos the pan-tilt camera. A feed-forward controller was also implemented to overcome instabilities and improve tracking robustness [9]. In that work, the tracking performance between proportional and feed-forward controllers were compared but no platform model was included in the controller.

This paper incorporates the platform dynamics into the feed-forward control scheme [4]. The system uses a $\alpha - \beta - \gamma$ filter described in [9], [4] and [6]. A condensation algorithm [5] is used for target detection. The non-linear controller improves performance and stability as underscored by experimental results.

0-7803-9098-9/05/$25.00 ©2005 AACC

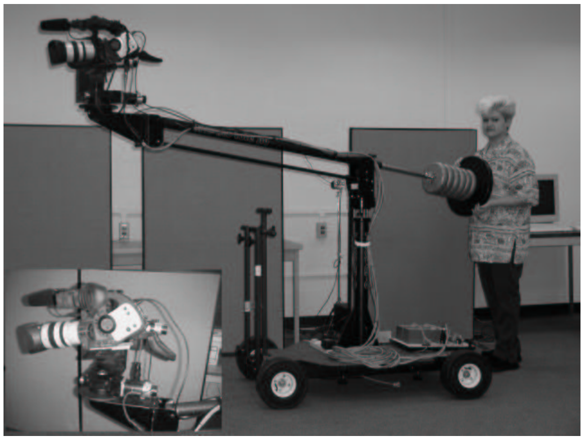

Figure 1: An operator can boom the arm horizontally and vertically to position the camera. The pan-tilt (lower left inset) head provides additional degrees-of-freedom.

Table 1: Types of motion for links.

Joint #	RB	JB	x	y	R_x	R_y	R_z
1		1	x	y			
2	1	2					ψ_b
3	2	3				θ_{bt1}	
4	2	4				θ_{bb1}	
5	3	5				θ_{bt2}	
6	4	5				θ_{bb2}	
7	5	6					ψ_c
8	6	7				θ_c	

Section 2 formulates and validates the boom model. Section 3 briefly describes the feed-forward controller. Design of the new controller is described in Section 4. Simulation and experimental results with this controller are described in Section 5. Conclusions are presented in Section 6.

2 Symbolic Modeling

Both the non-linear mathematical and simulation models of the boom were developed using *Mathematica* and *Tsi ProPac* [2] [3]. The former is in Poincaré equations enabling one to evaluate the properties of the boom to design either a linear or non-linear controller. The later is in the form of C-code that can be compiled as an S-function in Simulink. Together these models of the high-order and coupled boom dynamics facilitate the design and testing of the controller before its actual implementation on the real system. The boom shown in Figure 2 is comprised of 7 bodies and 8 joints. The bodies and joints are denoted by boxes and circles respectively. The degrees-of-freedom of the various joints are detailed in Table 1 while physical parameters are given in Table 2. They give the position or Euler angles of the *joint body*(JB) with respect to the *reference body*(RB). At the origin, which corresponds to stable equilibrium, the boom and the camera are perfectly aligned. One characteristic of the boom is that it always keeps the camera's base parallel to the floor. This is because bodies 3 and 4 are part of a four-bar linkage and are two constraints for the system. These can be seen in Equation 1. Thus the system effectively has only 16 degrees-of-freedom.

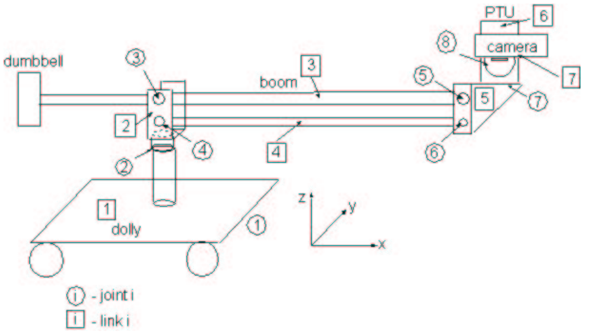

Figure 2: A number was assigned to every link and joint. Circled numbers represents joints while a number in a rectangle represent a link

$$\begin{aligned} \theta_{bb1} - \theta_{bt1} &= 0 \\ \theta_{bt1} + \theta_{bt2} &= 0 \end{aligned} \tag{1}$$

The inputs acting on the system are the torques $Q1$ (about y) and $Q2$ (about z) exerted by the operator, and the torques $Q3$ and $Q4$ applied by the pan and tilt motors of the camera, that is, $\mathbf{u} = \{Q1, Q2, Q3, Q4\}$. One pushes on the dumbbell counterweight at the end of body 3 to facilitate target tracking with the camera. In this analysis, it is assumed that the operator does not move the cart, although it is straightforward to incorporate that as well. The pan and tilt motors correspond to the rotations ψ_c and θ_c respectively.

Eventually the model is obtained in the form of Poincaré equations (see [2] and [3] for details)

$$\begin{aligned} \dot{\mathbf{q}} &= V(\mathbf{q})\mathbf{p} \\ M(\mathbf{q})\dot{\mathbf{p}} + C(\mathbf{q})\mathbf{p} + \mathbf{Q}(\mathbf{p}, \mathbf{q}, \mathbf{u}) &= \mathbf{0} \end{aligned} \tag{2}$$

The generalized coordinate vector $\mathbf{q}$, (see Table 1 for notation) is given by $\mathbf{q} = [x, y, \psi_b, \theta_{bt1}, \theta_{bt2}, \theta_{bb1}, \theta_{bb2}, \psi_c, \theta_c]^T$. The vector

$\mathbf{p}$ is the 7×1 vector of quasi-velocities given by $\mathbf{p} = [\Omega_{yc}, \Omega_{zc}, \Omega_{bb2}, \Omega_{bt2}, \Omega_{zb}, v_y, v_x]^T$. They are the quasi-velocities associated with joints 8, 7, 6, 5, 2 and double-joint 1 respectively. The first set of equations are the kinematics and the second are the dynamics of the system. A simulation model was generated and compiled into a Matlab dynamically linked library (DLL). Simulink simulations were compared with experimental data and show close correlation.

3 Feedforward Controller

As mentioned in [8], the boom-camera system under proportional control becomes unstable when tracking fast moving targets. The boom and the pan-tilt unit are redundant rotational DOF that, at high frequencies, can become 180 degrees out-of-phase. The net result is the boom and the pan-tilt unit rotations conflict, rather than cooperate, and tracking fails. To overcome such instabilities, a feed-forward controller can be designed which provides target motion estimation [4]. This controller is briefly described in this section.

When tracking a moving target, the target position and velocity are not known a priori. Still, in the image plane these parameters can be estimated. Considering the configuration of the system (given by encoders) the position and velocity can be estimated in real world coordinates. The feed-forward controller is based on this idea. As a feedback compensator, it uses a simple PID regulator [4]. Figure 3 depicts the feed-forward scheme. Here, $V(z)$ and $G_p(z)$ are the transfer functions for the vision system and the pan-tilt unit respectively. Together $G(z)$, $G_F(z)$ represent the feed-forward transfer functions. $D(z)$ is the feedback controller transfer function.

For a horizontally translating target, its centroid in the image plane is given by the relative angle between the camera and the target

$$^{i}X(z) = K_{lens}(X_t(z) - X_r(z)) \tag{3}$$

where $^{i}X(z)$ and $X_t(z)$ are the target position in the image plane and world frame respectively. $X_r(z)$ is the position of the point which is in focus (due to the booming and camera rotation) and K_{lens} is the lens zoom value. Rearranging this equation yields

$$\hat{X}_t(z) = \frac{^{i}\tilde{X}(z)}{K_{lens}} + X_r(z) \tag{4}$$

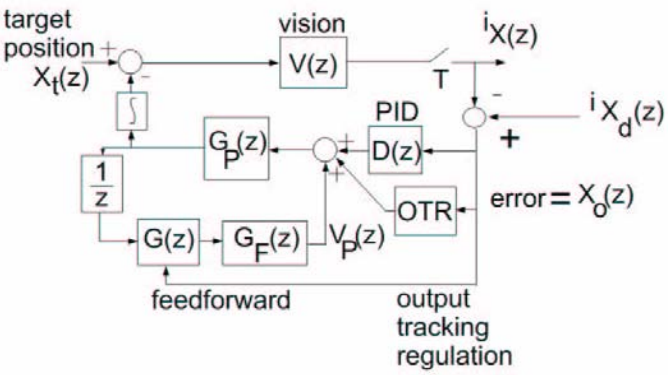

Figure 3: The feedforward controller with compensation. The output tracking regulation controller (OTR) is added to the scheme.

where $\hat{X}_t$ is the predicted target position.

The constant, K_{lens}, represents the ratio between the target dimensions in the image plane (in pixels) and in meters. K_{lens} was set to a constant value and assumes a pinhole camera model that maps the image plane and world coordinates. This constant was experimentally determined by comparing known lengths in world coordinates to their projections in the camera's image plane.

3.1 $\alpha - \beta - \gamma$ Filter

Predicting target velocity requires a tracking filter. Oftentimes a Kalman filter is used but this is computationally expensive. Since Kalman gains often converge to constants, the simpler $\alpha - \beta - \gamma$ tracking filter can be employed which tracks both position and velocity without steady-state errors [6].

Tracking via this method is a two step process. The first step is to predict target position and velocity for the next iteration

$$x_p(k+1) = x_s(k) + Tv_s(k) + T^2 a_s(k)/2 \tag{5}$$

$$v_p(k+1) = v_s(k) + Ta_s(k) \tag{6}$$

where T is the sample time and $x_p(k+1)$ and $v_p(k+1)$ are respectively the predictions for position and velocity at iteration $k+1$. $x_s(k)$, $v_s(k)$ and $a_s(s)$ are the corrected values of iteration k for position, velocity and acceleration respectively.

The second step is to make corrections

$$x_s(k) = x_p(k) + \alpha(x_o(k) - x_p(k)) \tag{7}$$

$$v_s(k) = v_p(k) + (\beta/T)(x_o(k) - x_p(k)) \tag{8}$$

$$a_s(k) = a_p(k-1) + (\gamma/2T^2)(x_o(k) - x_p(k)) \tag{9}$$

where $x_o(k)$ is the observed (sampled) position at iteration k. The appropriate selection of gains α, β and γ will determine the performance and stability of the filter.

In [9] an $\alpha - \beta - \gamma$ filter was implemented to predict target velocity in the image plane. This velocity was then used in the feed-forward algorithm as shown in Figure 3.

4 Tracking Regulation Controller

This work investigated the effectiveness and advantages of the controller implemented as a regulator with disturbance rejection properties. This approach guarantees regulation of the desired variables, while simultaneously stabilizing the system and rejecting exogenous disturbances. As a first step, a linear controller was designed to regulate only the pan motion.

The linearized equations are recast as

$$\begin{aligned} \dot{x} &= Ax + Pw + Bu \\ \dot{w} &= Sx \\ e &= Cx + Qw \end{aligned} \tag{10}$$

The regulator problem is solvable if and only if the linear matrix equations 11 are solved by Π and Γ [1].

$$\begin{aligned} \Pi S &= A\Pi + P + B\Gamma \\ 0 &= C\Pi + Q \end{aligned} \tag{11}$$

A regulating controller can then be constructed as

$$u = \Gamma w + K(x - \Pi w) \tag{12}$$

where K is chosen so that the matrix $(A + BK)$ has desired eigenvalues. These eigenvalues determine the quality of the response.

The model has the transfer function

$$\frac{\theta(s)}{V_a(s)} = \frac{0.01175}{1.3s^2 + 32s} \tag{13}$$

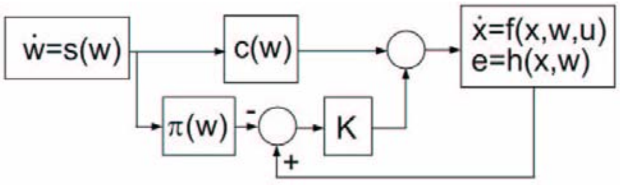

Figure 4: The Output Tracking Regulation Controller as it was implemented.

where the output is the camera angle. In this case, the state space description of the system is given by matrices A, B and C

$$A = \begin{vmatrix} -24.61 & 0 \\ 1 & 0 \end{vmatrix}$$

$$B = \begin{vmatrix} 0.0088 \\ 0 \end{vmatrix}$$

and

$$C = \begin{vmatrix} 0 & 1 \end{vmatrix}$$

From Equation 11

$$\Pi = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}$$

while

$$\Gamma = \begin{vmatrix} -113.6 & 2796.6 \end{vmatrix}$$

The matrix K was

$$K = \begin{vmatrix} -10000 & -380 \end{vmatrix}$$

5 Simulations and Experiments

Prior to implementation, the new controller was simulated using Matlab-Simulink. A $1 rad/sec$ sinusoidal reference signal was applied to the controller. Both, the reference and the output of the system were plotted on the same axes frame and showed close correspondence. With confidence in the model, the output tracking regulation controller was added to the Visual C program. To assess the increase of performance due to the vision system an experiment was set up with the Mitsubishi robot. The robot moved the target in a figure-eight trajectory for 60 *seconds*. Both experienced and inexperienced operators were asked to boom the camera along a predefined path. Each operator boomed twiced, first using the vision system and then, the camera was rotated by the operator using a joystick. The objective was to keep the target in the camera's field-of-view while both the target and

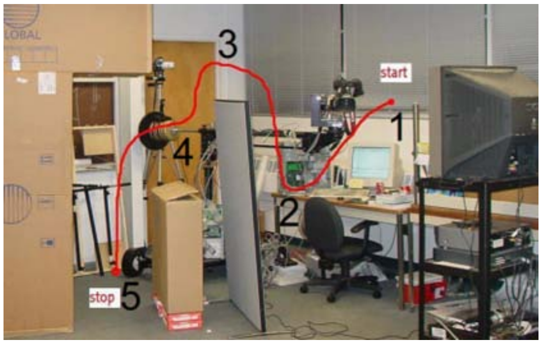

Figure 5: Camera path used by the operator.

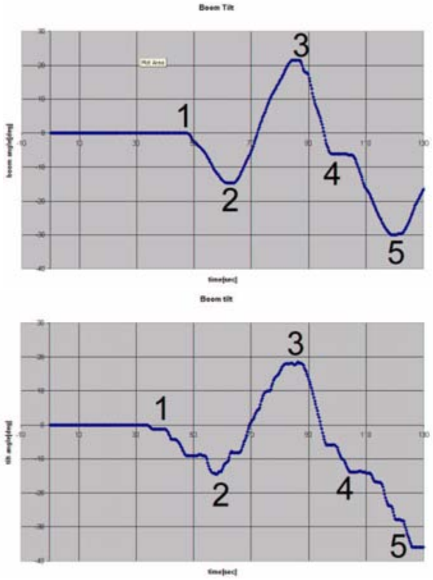

Figure 6: Boom tilt angle in case of unexperienced operator with (top) and without vision (bottom).

boom moved. The boom was restricted to move on a certain path. The booming path can be seen in Figure 5. In Figure 5 several positions are marked along the booming path using numbers. Boom tilt angle can be seen in Figure 6. Here, the top figure represents the tilt angle when the vision system was used. The bottom figure, represents the same angle but this time, without visual-servoing. It can be seen that in second case, booming was not smooth.

The experiment was videotaped. Sequential images can be seen in Figure 7. The images wer taken when the camera was in one of the positions marked in Figure 5. With visual-servoing, the target was never lost as seen Figure 7 (top row).

Table 2: Boom links, masses and moments of inertia.

Object	Mass *kg*	Moment of Inertia $kg\ m^2$
Dolly (link1)	25	$I_{xx} = 2.48$, $I_{yy} = 0.97$, $I_{zz} = 3.465$
Link 2	0.6254	$I_{xx} = 0.000907$ $I_{yy} = 0.000907$, $I_{zz} = 0.00181$
Boom (link 3)	29.5	$I_{xx} = 0$ $I_{yy} = 16.904$, $I_{zz} = 16.904$
Link 4	0.879	$I_{xx} = 0$ $I_{yy} = 0.02379$, $I_{zz} = 0.02379$
Link 5	3.624	$I_{xx} = 0.08204$ $I_{yy} = 0.00119$, $I_{zz} = 0.00701$
PTU (link 6)	12.684	$I_{xx} = 0.276$ $I_{yy} = 0.234$, $I_{zz} = 0.0690$
Camera (link 7)	0.185	$I_{xx} = 0$ $I_{yy} = 1.33\dot{1}0^{-5}$, $I_{zz} = 1.33\dot{1}0^{-5}$

6 Conclusions and Future Work

This paper integrates visual-servoing to augment the tracking performance of camera teleoperators. By reducing the number of DOF that needs to be manually manipulated, the operator can concentrate on coarse camera motion. Using a broadcast boom camera as an experimental platform, the dynamics of the boom and pan-tilt-unit were derived and validated experimentally. A new controller was added to the feed-forward scheme and was tested experimentally. Both experienced and unexperienced operator performance with and without computer vision were assessed. Experiments revealed that the vision system helps the operator keep the target centered in the camera's field-of-view despite challenging target motions and inexperience.

References

[1] Isidori A. "Nonlinear Control Systems. 3rd ed." *Springer Verlag 1995*

[2] Kwatny H.G., Blankenship G.L."Nonlinear Control and Analytical Mechanics: a computational

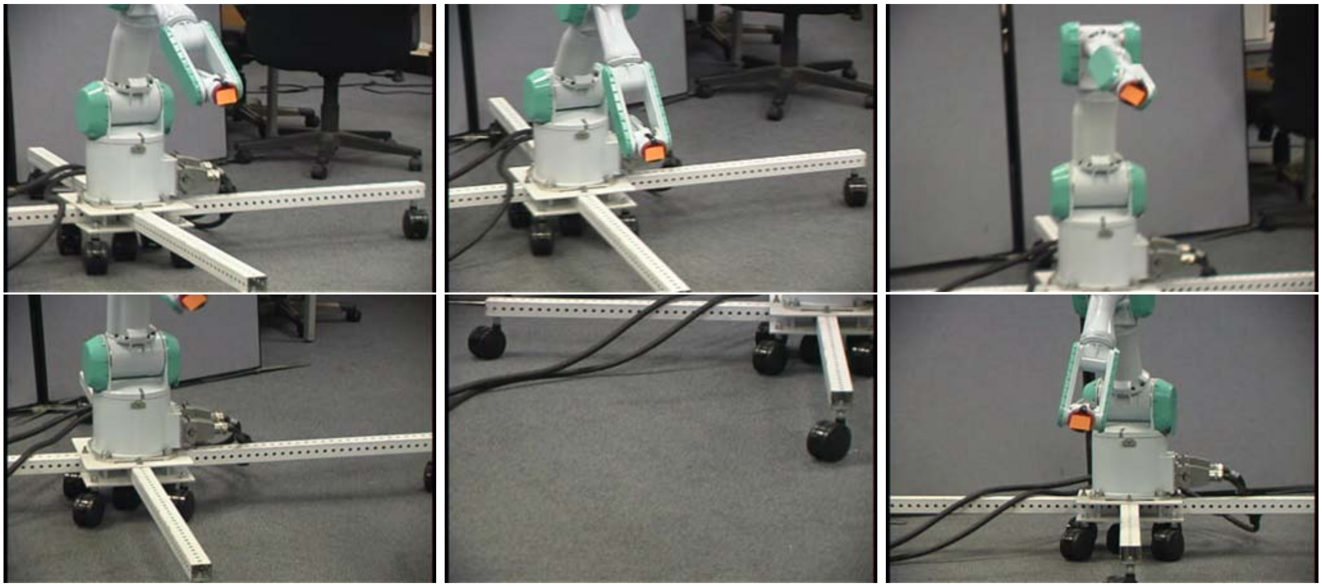

Figure 7: The boom is moved as shown in Figure 5. Augmented with visually servoed control, the unexperienced operator can keep the target centered in the camera's field-of-view (top row). Without visual-servoing, the operator often fails (bottom row)

approach" *Boston: Birkhauser, 2000*

[3] Kwatny H.G., Blankenship G.L."Symbolic construction of models for multibody dynamics" *IEEE Trans on Robotics and Automation* vol. 11 no. 2 pp. 271-281 April 1995.

[4] Corke P.I., Good M.C."Dynamic Effects in Visual Closed-Loop Systems" *IEEE Trans on Robotics and Automation* vol. 12 no. 5 pp. 671-683 Oct 1996

[5] Isard, M., Blake, A., "CONDENSATION – Conditional Density Propagation for Visual Tracking", *Int. J. Computer Vision*, V29 N1 pp. 5-28, 1998.

[6] Kalata, P.R., Murphy, K.M., '$\alpha-\beta$ Target Tracking with Track Rate Variations", *Proceedings of the Twenty-Ninth Southeastern Symposium on System Theory*, pp. 70-74, Mar 1997.

[7] Oh, P.Y., Allen, P.K., "Visual Servoing by Partitioning Degrees of Freedom", *IEEE Trans on Robotics Automation* V17 N1, pp. 1-17, Feb 2001.

[8] Stanciu, R., Oh, P.Y., "Designing Visually Servoed Tracking to Augment Camera Teleoperators" *IEEE Intelligent Robots and System (IROS)*, Lausanne, V1, pp. 342-347, 2002.

[9] Stanciu, R., Oh, P.Y., "Feedforward Control for Human-in-the-loop Camera Systems" *IEEE Int. Conf. on Robotics and Automation (ICRA)*, V1, pp. 1-6, New Orleans, LA, Apr 2004.